Evaluating the Importance of Time Series Features in Eye Movement Traces for Cognitive Load Differentiation

Aryan Lodha
Supervised by: Dr Gabriella Miles

Abstract

Cognitive load understanding and detection are important in a variety of domains, including education, cognitive neuroscience, and user experience design. This project aims to determine the extent to which certain temporal characteristics visible in eye movement data can help distinguish between low and high cognitive load conditions. Eye movements—comprising saccades, fixations, and blinks—offer insight into mental effort and task difficulty.

Using the *EM-COGLOAD* dataset—eye movement recordings of participants watching videos under varying cognitive loads—we will implement and evaluate a range of deep learning models and explainability techniques to detect load-specific signals. Our approach includes preprocessing raw gaze coordinates, deriving features such as velocity and blink rate, training deep neural networks, and interpreting results using GradCAM++ and SHAP.

The outcome will be a reliable and interpretable model capable of identifying gaze-based indicators of mental effort. This research may contribute to the development of future tools for adaptive e-learning and cognitive state monitoring in healthcare.

Ethics Statement: This project fits within the scope of ethics application 97842, as reviewed by my supervisor, Dr Gabriella Miles. I have completed the ethics test on Blackboard with a score of **12/12**.

1 Background and Motivation

Cognitive load is a critical determinant of the extent to which individuals are able to **process**, **store**, and **recall** information in real-time performance. It is a foremost concern in applications such as **education**, **medicine**, and **human-computer interaction**, where too much mental effort can compromise decision-making and learning performance. Traditional methods for assessing cognitive load—such as **self-reports**, **NASA-TLX questionnaires**, or **post-task interviews**—are retrospective and subjective in nature [1]. Thus, they are not very effective at facilitating real-time adaptation or unobtrusive monitoring.

This limitation has created greater interest in **on-line, passive** measures of cognitive load. One particularly promising area is the study of **eye movement activity**. Human eye movement has certain characteristics—e.g., **saccades**, **fixations**, and **blinks**—in which cognitive processes find expression. Higher cognitive loads, for example, tend to be accompanied by **higher saccadic frequencies**, **shorter fixation durations**, and **reduced blinking** [1]. Information obtained through **eye tracking** offers a continuous signal that can be recorded in a non-disruptive manner. Prior studies have also established the viability of using **instruction-free**, **naturalistic eye-tracking** to assess cognition in applied settings, including **medical diagnostics** [2].

This project builds on these findings to investigate whether **eye movement signals** can reliably differentiate between high and low cognitive load using machine learning. We focus our analysis on the ***EM-COGLOAD*** dataset [3][1], which includes participants watching videos under both passive and memory-task conditions. Trial suffixes such as _0 and _2 indicate the same video viewed under low and high load respectively, enabling structured comparison.

Our primary hypothesis is that temporal and statistical features—such as **gaze velocity**, **blink rate**, and **fixation entropy**—will diverge across conditions. We will extract these features and train **deep learning models** on both raw and engineered time-series inputs. For model transparency, we will apply **Explainable AI (XAI)** tools, including **Grad-CAM++** [4] and **SHAP** [5], to interpret predictions.

Ultimately, our goal is to develop a **reliable**, **interpretable**, and **practical** framework for cognitive load detection via eye-tracking—supporting future use in **adaptive learning systems**, **workload-aware interfaces**, and **clinical evaluation**.

[1] `https://osf.io/df3rw/`

2 Project Plan

This project is divided into six key phases spanning approximately ten weeks over the summer. While each phase has a primary focus, some overlap is expected as analysis, modelling, and writing often develop in parallel. A detailed timeline and risk strategy are provided in Appendix A and Appendix B.

Phase 1 – Data Setup and Exploration

- Load and inspect the EM-COGLOAD dataset [3][2].
- Review trial structure and file conventions.
- Identify and handle any missing or inconsistent entries.

Phase 2 – Feature Engineering

- Preprocess raw gaze coordinates and normalize time series.
- Derive key features: gaze velocity, blink rate, fixation entropy, etc.
- Conduct initial exploratory analysis to evaluate feature quality.

Phase 3 – Baseline Modeling

- Train simple models (logistic regression, random forests) on engineered features.
- Implement a 1D CNN to learn patterns directly from raw time series.
- Compare results across model types for performance benchmarks.

Phase 4 – Model Refinement and Explainability

- Tune model parameters and architecture based on validation results.
- Use Grad-CAM++ [4] and SHAP [5] to interpret feature and temporal importance.
- Generate visualizations to explain model behaviour.

Phase 5 – Results Analysis

- Compare models in terms of accuracy and interpretability.
- Summarize findings and identify gaze features most linked to cognitive load.
- Refine visuals and structure insights for reporting.

Phase 6 – Writing and Submission

- Complete and polish the dissertation.
- Review with supervisor and implement feedback.
- Submit final version by the university deadline.

While each step is planned sequentially, adjustments will be made based on progress and results. Iteration is expected, particularly between modelling and feature development.

[2] `https://osf.io/df3rw/`

References

[1] C.-M. Chen, C.-C. Lin, H.-C. Lin, and J.-T. Chien, “Eye tracking metrics to predict perceived task difficulty in human-robot interaction,” *IEEE Access*, vol. 8, pp. 69 261–69 271, 2020.

[2] K. Mengoudi, D. Ravi, K. X. X. Yong, S. Primativo, I. M. Pavisic, E. Brotherhood, K. Lu, J. M. Schott, S. J. Crutch, and D. C. Alexander, “Augmenting dementia cognitive assessment with instruction-less eye-tracking tests,” *IEEE Journal of Biomedical and Health Informatics*, vol. 24, no. 11, pp. 3066–3077, 2020.

[3] G. Miles, M. Smith, N. Zook, and W. Zhang, “Em-cogload: An investigation into age and cognitive load detection using eye tracking and deep learning,” *Computational and Structural Biotechnology Journal*, vol. 24, pp. 264–280, 2024.

[4] A. Chattopadhyay, A. Sarkar, P. Howlader, and V. N. Balasubramanian, “Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks,” in *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2018, pp. 839–847.

[5] S. M. Lundberg and S.-I. Lee, “A unified approach to interpreting model predictions,” *Advances in Neural Information Processing Systems*, vol. 30, pp. 4765–4774, 2017. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2017/file/8a20a8621978632d76c43dfd28b67767-Paper.pdf

A Project Timeline

The project is structured around six primary phases, mapped across ten weeks during the summer period. While the sequence is designed to be linear, some overlap is natural—especially between modelling and analysis phases. The breakdown below offers a practical and flexible schedule that allows for iteration and refinement.

- **Week 1:** Initial setup and dataset familiarisation. Understand the file structure, trial conditions, and data completeness. Generate summary statistics (e.g., trace lengths, blink counts) and spot-check examples visually.
- **Weeks 2–3:** Feature extraction and signal preparation. Clean and normalize the gaze data, derive velocity, blink frequency, and entropy metrics, and visualise differences across cognitive load conditions.
- **Weeks 4–5:** Baseline model development. Implement logistic regression and random forest classifiers on the engineered features. Simultaneously, build a simple 1D CNN to model raw time series. Evaluate and compare these approaches using cross-validation.
- **Weeks 6–7:** Deep learning refinement and interpretability. Improve CNN architecture and training stability. Apply Grad-CAM++ and SHAP to identify which features and time points most influence predictions. Validate findings against cognitive load theory and previous work [2].
- **Weeks 8–9:** Result synthesis and insight generation. Compile key plots, evaluate model generalisability, and write analytical summaries of what features most reliably distinguish low vs high cognitive load.
- **Week 10:** Final writing, review, and submission. Prepare the full dissertation. Include figures, tables, and citations. Submit following university guidelines before the August deadline.

B Risk Management

The table below summarizes potential project risks along with their likelihood, impact, and mitigation strategies. The table is placed on its own page for clarity and proper formatting.

Table 1: Risk Assessment and Mitigation

Risk	**Likelihood**	**Impact**	**Mitigation Strategy**
Dataset issues (e.g., missing or corrupted trials)	Medium	Medium	Identify problems early during Phase 1; filter or exclude missing data and document decisions clearly.
Deep learning model underperforms or overfits	Medium	Medium	Begin with simple models first; use cross-validation and dropout layers; have fallback models ready.
Explainability tools (Grad-CAM++, SHAP) provide weak or noisy outputs	Medium	Medium	Use multiple XAI techniques and visualizations; revise model or focus on interpretable feature-based models if needed.
Time pressure during final phase	Medium	High	Begin writing early (during Weeks 6–7); reserve Week 10 solely for review and editing.
Technical issues (hardware, software, version control problems)	Low	High	Use GitHub regularly; keep local and cloud backups; maintain access to secondary computing resources.